

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2001-168762
(P2001-168762A)

(43) 公開日 平成13年6月22日 (2001.6.22)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード*(参考)
H 0 4 B	1/707	H 0 3 M 7/30	A 5 J 0 6 4
H 0 3 M	7/30	H 0 4 L 7/00	D 5 K 0 2 2
H 0 4 B	7/26	H 0 4 J 13/00	D 5 K 0 4 7
H 0 4 L	7/00	H 0 4 B 7/26	N 5 K 0 6 7

審査請求 未請求 請求項の数11 O L (全 13 頁)

(21) 出願番号 特願平11-345903

(22) 出願日 平成11年12月6日 (1999.12.6)

(71) 出願人 000005821

松下電器産業株式会社
大阪府門真市大字門真1006番地

(72) 発明者 栗原 直之

大阪府門真市大字門真1006番地 松下電器
産業株式会社内

(74) 代理人 100105050

弁理士 鷲田 公一

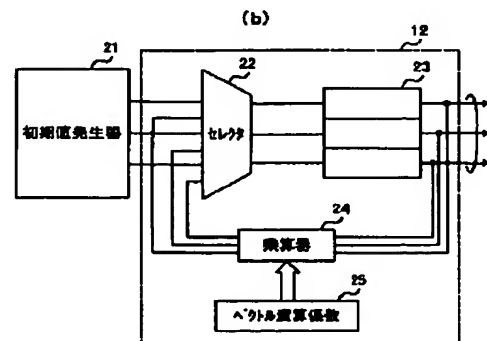
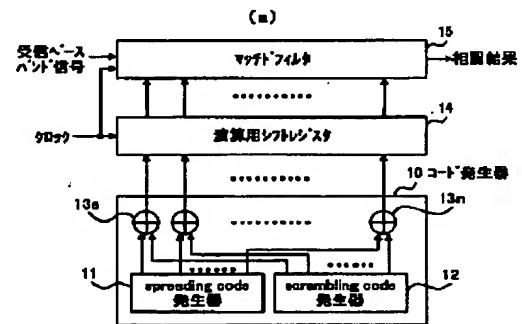
Fターム(参考) 5J064 AA03 BA16 BC01 BC03 BC04
BC29 BD02
5K022 EE02 EE25
5K047 AA02 GG34 HH15 JJ06 MM27
MM29 MM53
5K067 CC10 DD25 EE10 HH23

(54) 【発明の名称】 符号発生器および初期同期装置

(57) 【要約】

【課題】 回路の複雑化を防止しつつ、移動体通信における初期同期の獲得を高速に行なうこと。

【解決手段】 コード発生器10において、スクランプリング・コードとスプレッディング・コードを、それぞれ並列に発生させる。スクランプリング・コードは、ベクトル演算を工夫することで、パラレル生成を可能とする。スプレッディング・コードについては、ワイヤードROMを使用した簡単な回路でもって、アダマール符号や階層化直交符号をパラレルに生成する。



【特許請求の範囲】

【請求項１】 所定ビット数のアダマール符号もしくは階層化直交符号をパラレルに発生させる符号発生器であって、

前記所定ビット数の参照用データ列が複数登録されているテーブルと、

このテーブル内に登録されている、前記参照用データ列の各々に対応して設けられ、各参照用データ列を構成する各ビットのデータと、発生させたいアダマール符号もしくは階層化直交符号を指定するデータとのアンドをとって、その結果をパラレルに出力する、複数のアンド演算回路と、

これらの複数のアンド演算回路の各々からパラレルに出力される前記所定ビット数のデータの各ビットについて排他的論理和演算を行い、前記所定ビット数のアダマール符号もしくは階層化直交符号をパラレルに出力する排他的論理和演算回路と、を有する符号発生器。

【請求項２】 前記テーブルおよび複数のアンド演算回路は、ハードワイヤードロジックにより、予めプログラムされたROMであることを特徴とする請求項１記載の符号発生器。

【請求項３】 請求項１または請求項２記載の符号発生器からパラレルに出力される符号と、ベクトル行列演算とシフトレジスタを用いて生成され、パラレルに出力される符号との排他的論理和演算により、CDMA通信方式の初期同期獲得に必要な拡散符号をパラレルに生成することを特徴とする符号発生器。

【請求項４】 受信ベースバンド信号を蓄積する蓄積手段と、演算用符号をパラレルに発生する符号発生手段と、この符号発生手段からパラレルに発生する前記演算用符号を一時的に格納する格納手段と、前記蓄積手段に蓄積された前記受信ベースバンド信号と前記格納手段に格納された演算用拡散符号との相関を検出する相関検出手段と、を有することを特徴とする初期同期装置。

【請求項５】 前記格納手段は、格納する演算用拡散符号をパラレルに連続サイクルで切り換え、これにより、前記相関検出手段において、前記蓄積手段に蓄積された受信ベースバンド信号との相関検出が連続的に行なわれることを特徴とする請求項４記載の初期同期装置。

【請求項６】 前記符号発生手段は、01、0011といった0と1の繰り返しの参照用データとアダマール符号列を指定する番号とを各々論理積をとる論理積演算回路と、前記論理積演算回路の演算結果の排他的論理和をとる排他的論理和演算回路と、を有し、パラレルに連続サイクルでアダマール符号列を発生させることを特徴とする請求項４または請求項５記載の初期同期装置。

【請求項７】 前記符号発生手段は、01、0011といった0と1の繰り返しの参照用データと階層化直交符号列を指定する番号との論理積をとる論理積演算回路と、前記論理積演算回路の演算結果の排他的論理和をとる

排他的論理和演算回路と、を有し、パラレルに連続サイクルで階層化直交符号列を発生させることを特徴とする請求項４または請求項５記載の初期同期装置。

【請求項８】 前記符号発生手段は、パラレルにゴールド符号を発生させるゴールド符号発生回路と階層化直交符号列をパラレルに発生する階層化直交符号発生回路とを有し、ゴールド符号列と階層化直交符号列との排他的論理和をとることにより、パラレルにWCDMAシステムにおける第３段階の初期同期をとるための拡散符号列を発生させることを特徴とする請求項４または請求項５記載の初期同期装置。

【請求項９】 受信ベースバンド信号を受信し、請求項４～請求項８のいずれかに記載の初期同期装置に入力して拡散符号列との相関検出を行い、相関検出結果に基づいて初期同期獲得を行うことを特徴とするCDMA受信装置。

【請求項１０】 請求項９記載のCDMA受信装置を備え、前記CDMA受信装置にCDMA無線通信で受信した受信ベースバンド信号を入力して初期同期獲得し、獲得した同期タイミングに基づいて通信制御を行うことを特徴とする移動体通信端末装置。

【請求項１１】 請求項９記載のCDMA受信装置を備え、前記CDMA受信装置にCDMA無線通信で受信した受信ベースバンド信号を入力して同期を獲得し、獲得した同期タイミングに基づいて通信制御を行うことを特徴とする移動体通信基地局装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、スペクトラル拡散通信方式で同期獲得に用いられる、符号発生器および初期同期装置に関する。

【0002】

【従来の技術】スペクトラム通信では、通信チャンネル毎に異なる拡散符号を用いて拡散変調を行い送信する。受信側では、送信側で用いた拡散符号と同じ符号で逆拡散を行なう必要がある。その為、拡散符号の検出及びタイミングを初期同期において確立しなければならない。

【0003】この初期同期方法として、３段階初期同期方法があり、１段階でチップ同期の確立を行い、２段階でスプレッディング符号グループの同定及びフレームタイミングの確立を行い、３段階でスプレッディング符号を同定する。

【0004】この２段階及び３段階においては、複数の種類の拡散符号の中から１種類をいかに早く同定するかが、システム高性能化への課題である。マッチドフィルタを用いる場合は、受信ベースバンド信号を蓄積し、同じ受信ベースバンド信号に対し、複数の拡散符号列との相関検出を行い、拡散符号の検出及びタイミングの確立を行う。

【0005】従来のマッチドフィルタを用いた初期同期

装置の回路構成について図11を用いて説明する。図11の初期同期装置は、マッチドフィルタ1001と、拡散符号発生器1002と、演算用の拡散符号を蓄積する演算用レジスタ1003とで構成されている。

【0006】同図に示すマッチドフィルタ1001は、FIRデジタルフィルタを用いた4倍拡散のマッチドフィルタの例である。このマッチドフィルタ1001は、シフトレジスタ1008と、シフトレジスタ1008に蓄えられた受信ベースバンド信号と拡散符号列との逆拡散演算を実行する乗算器群1009～1012と、乗算結果を加算する加算器1013とを備える。

【0007】図示されるように、シフトレジスタ1008は、直列接続された4段のフリップフロップ1004～1007で構成される。一段目のフリップフロップ1004には、受信ベースバンド信号が入力される。また、フリップフロップ1004～1007の各々には、動作クロックが平行に与えられ、この動作クロックに同期して、記憶データが1段ずつシフトされる。

【0008】拡散符号発生器1002は、拡散符号列を1符号ずつシリアルに発生し、4クロック（拡散率分のクロック数）で拡散符号列分の拡散符号を発生させる。

【0009】演算用レジスタ1003は、直列接続された複数のフリップフロップ1014～1017と、直列接続された複数のフリップフロップ1018～1021とを具備する。

【0010】フリップフロップ1014には、拡散符号発生器1002から出力される拡散符号がシリアルに入力される。フリップフロップ1014～1017には、動作クロックが平行に入力されており、クロックに同期してデータが右側に一段ずつシフトされる。フリップフロップ1014～1017に拡散符号列が格納されたあと、フリップフロップ1014～1017の値（記憶されている値）を、フリップフロップ1018～1021に平行にロードする。なお、フリップフロップ1018～1021にも、動作クロックが、平行に与えられる。

【0011】次に、初期同期装置の動作について説明する。

【0012】受信ベースバンド信号は、クロックに同期した6ビットの信号である。デジタル信号は、フリップフロップ1004に入力されたのち、フリップフロップ1007に向けてクロックに同期してシフトされる。4クロック後には、フリップフロップ群1004～1007にデータが蓄積され、フリップフロップ1004～1007に平行に与えられていたクロックは停止される。これにより、フリップフロップ1004～1007に蓄積された受信ベースバンド信号の値は保持される。

【0013】次に、拡散符号発生器よりシリアルに出力された拡散符号列は、フリップフロップ1014に入力

されたのち、フリップフロップ1017に向けてクロックに同期してシフトされる。4クロック後には、フリップフロップ1014～1017に、拡散符号列は格納される。次のクロックでは、フリップフロップ1014～1017に格納された拡散符号列がフリップフロップ1018～1021に平行にロードされる。以上により、受信ベースバンド信号との相関検出を行なう拡散符号列がセットされる。

【0014】乗算器1009～1012は、6ビット×1ビットの乗算器であり、7ビットの信号を出力する。また、加算器1013は、7ビット+7ビット+7ビット+7ビットの加算器であり、9ビットの信号を出力する。

【0015】乗算器1009では、フリップフロップ1004（6ビット）の出力信号とフリップフロップ1018の出力信号との乗算が行われる。乗算器1010～1012では、フリップフロップ1005～1007とフリップフロップ1019～1021との乗算がそれぞれ行われる。加算器1013では、乗算器1009～1012の出力の加算が行われる。すなわち、乗算器1009～1012および加算器1013を用いて、受信ベースバンド信号と拡散符号発生器1002で発生する拡散符号列との相関を検出する。

【0016】次に、拡散符号列の切り換えの動作について説明する。

【0017】次の相関演算を行なう、拡散符号列が拡散符号発生器1002よりシリアルに出力されると、クロックに同期してフリップフロップ1018～1021に5クロックで格納されるため、これに対応して、拡散符号列の切り換えも、5クロック毎に行なわれる。拡散符号の切り換えが完了後、シフトレジスタ1008に蓄えられている受信ベースバンド信号と切り換え後の拡散符号列との相関検出が行なわれる。

【0018】同様に、複数の種類の拡散符号列の切り換えを連続して行うとすると、相関検出を行なおうとしている拡散符号列の種類が10種類であれば、10種類の拡散符号列に対し50クロックで相関検出が行なわれる。本動作は、受信ベースバンド信号の拡散率が変わっても同様な動作で行なうことが可能である。

【0019】

【発明が解決しようとする課題】上述の従来の初期同期装置の構成では、演算用レジスタへの拡散符号の格納をシリアルに行なっているため、格納に時間がかかり、拡散符号を高速に切替えることが困難である。このため、拡散符号の検出及びタイミングの確立をする時間の高速化が容易でなく、さらには初期同期の確立に時間がかかりすぎるとマスタークロックの位相のずれなどにより、初期同期を行なっている間に送受信のタイミングがずれてしまい、初期同期のやり直しが発生するなどして、さらに時間がかかってしまうような事態も起こり得る。

【００２０】本発明は、このような従来の問題を解決するためになされたものであり、複数の拡散符号との演算を高速に処理し、高速に初期同期を確立することを目的とする。

【００２１】

【課題を解決するための手段】本発明では、初期同期を行なう際に必要な拡散符号であるアダマール符号、階層化直交符号、ゴールド符号の符号列をパラレルで出力する発生器を実現することにより、演算用レジスタへの格納をパラレルかつ連続サイクルで切り換えを実現する。

【００２２】パラレルかつ連続サイクルで切り換えを行えるので、シリアルに行なっていた場合に比べて、拡散符号列の発生及び格納スピードが格段に向上する。よって、従来構成に比べ、複数の種類の拡散符号との相関検出の高速化が図られ、初期同期の獲得を早期に行なうことができる。

【００２３】

【発明の実施の形態】本発明の符号発生器の一態様では、所定ビット数の参照用データ列が複数登録されているテーブルと、このテーブル内に登録されている、前記参照用データ列の各々に対応して設けられ、各参照用データ列を構成する各ビットのデータと、発生させたいアダマール符号もしくは階層化直交符号を指定するデータとのアンドをとって、その結果をパラレルに出力する、複数のアンド演算回路と、これらの複数のアンド演算回路の各々からパラレルに出力される前記所定ビット数のデータの各ビットについて排他的論理和演算を行い、前記所定ビット数のアダマール符号もしくは階層化直交符号をパラレルに出力する。

【００２４】テーブルと簡単な構成の演算回路とを用いて、アダマール符号もしくは階層化直交符号をパラレルに発生させるものであり、これにより、符号の発生速度が格段に向上する。

【００２５】また、他の態様では、前記テーブルおよび複数のアンド演算回路を、ハードワイヤードロジックにより、予めプログラムされたROMで構成する。

【００２６】また、他の態様では、ベクトル行列演算とシフトレジスタを用いて生成され、パラレルに出力される符号と、パラレル出力のアダマール符号等との排他的論理和演算により、CDMA通信方式の初期同期獲得に必要な拡散符号をパラレルに生成する。

【００２７】また、初期同期装置の一態様は、受信ベースバンド信号を蓄積する蓄積手段と、演算用符号をパラレルに発生する符号発生手段と、この符号発生手段からパラレルに発生する前記演算用符号を一時的に格納する格納手段と、前記蓄積手段に蓄積された前記受信ベースバンド信号と前記格納手段に格納された演算用拡散符号との相関を検出する相関検出手段と、を有する。この構成によれば、パラレルに拡散符号を格納手段に格納でき、相関検出の高速化が実現される。

【００２８】また、他の態様では、前記格納手段は、格納する演算用拡散符号をパラレルに連続サイクルで切り換え、これにより、前記相関検出手段において、前記蓄積手段に蓄積された受信ベースバンド信号との相関検出が連続的に行なわれる。連続サイクルで複数の種類の拡散符号との相関検出を行なえるため、拡散符号の検出のさらなる高速化が実現される。

【００２９】また、他の態様では、前記符号発生手段は、０１、００１１といった０と１の繰り返しの参照用データとアダマール符号列を指定する番号とを各々論理積をとる論理積演算回路と、前記論理積演算回路の演算結果の排他的論理和をとる排他的論理和演算回路と、を有し、パラレルに連続サイクルでアダマール符号列を発生させる。この構成によれば、初期同期の２段階処理に必要なアダマール符号列をパラレルにかつ連続サイクルで発生することを実現できるため、複数の種類の相関検出を高速に実現でき、２段階処理のスプレッディング符号グループの同定及びフレームタイミングの検出を高速に行なえる。

【００３０】また、他の態様では、前記符号発生手段は、０１、００１１といった０と１の繰り返しの参照用データと階層化直交符号列を指定する番号との論理積をとる論理積演算回路と、前記論理積演算回路の演算結果の排他的論理和をとる排他的論理和演算回路と、を有し、パラレルに連続サイクルで階層化直交符号列を発生させる。

【００３１】また、他の態様では、前記符号発生手段は、パラレルにゴールド符号を発生させるゴールド符号発生回路と階層化直交符号列をパラレルに発生する階層化直交符号発生回路とを有し、ゴールド符号列と階層化直交符号列との排他的論理和をとることにより、パラレルにWCDMAシステムにおける第３段階の初期同期をとるための拡散符号列を発生させる。この構成によれば、初期同期の３段階処理に必要な第３段階用符号列をパラレルに発生する事を実現できるため、複数の種類の相関検出を高速に実現でき、３段階処理のスプレッディング符号の同定を高速に行なえる。

【００３２】また、本発明の符号発生器および初期同期装置を用いて、CDMA受信装置、移動体通信端末装置、基地局装置を構成すると、きわめて高速に同期を獲得することができる。

【００３３】（実施の形態１）図１（a）は、本発明の実施の形態１に係る初期同期装置の構成を示すブロック図である。

【００３４】WCDMAシステムにおける同期獲得方法において、第２段階処理および第３段階処理では、同じタイミングで受信した信号に対して、任意のスクランプリング・コードとスプレッディング・コードとの任意の組み合わせで生成する複数の種類のコードとの相関検出を行なうことが必要である。本実施の形態の装置は、回

路規模を増やすことなく、初期同期の獲得を格段に高速化するものである。

【0035】図示されるように、初期同期装置は、コード発生器10と、演算用シフトレジスタ14と、マッチドフィルタ15とで構成される。コード発生器10からパラレルに発生する拡散コードは、演算用レジスタ14にパラレルにラッチされた後、パラレルに出力される。そして、マッチドフィルタ15にラッチされている受信ベースバンド信号と乗算され、相関結果がマッチドフィルタ15から出力される。

【0036】本実施の形態の特徴は、コード発生器10において、スプレッディング・コード (spreading code) および、スクランブリング・コード (scrambling code) を、共にパラレルに発生させ、加算器13a~13nで同時に演算していることである。これにより、1シンボル長の拡散コードを、演算用シフトレジスタ14に1サイクルで格納することができる。

【0037】すなわち、複数種類のコードの発生を、(必要なコード数×1) サイクルで行なうことが可能となる。シリアルにコードを発生させていた従来に比べ、相関検出に要する時間を、1/拡散率に短縮することができる。このため、複数のコード発生器を個別に用意することなく、一つのコード発生器を用いて、高速に初期同期を獲得することができる。このことは、回路規模の増大を抑制すると共に、消費電力の増大を抑えて携帯電話の長時間使用にも資するものである。よって、回路規模を増大させることなく、初期同期装置の高性能化を達成することができる。

【0038】スクランブリング・コード (scrambling code) としては、M系列符号あるいはM系列符号を基本とした符号 (例えば、Gold符号) を用いる。一方、スプレッディング・コード (spreading code) には、アダマール符号や階層化直交符号を用いる。

【0039】スクランブリング・コード発生器12は、例えば、図1(b)に示すような構成となっており、シフトレジスタ23の各段(図1(b)では、3段)からパラレルに符号を出力させる。

【0040】一方、パラレルに出力された符号に、乗算器24にて、ベクトル演算係数保持回路25から出力されるベクトル演算係数を乗算して、シフトレジスタ23において3ビット分だけ一挙にシフトした状態のデータをパラレルに発生させる。このデータは、セレクタ22を介してシフトレジスタ23の各段にパラレルに格納される。このような動作を繰り返して、1サイクルで複数のスクランブリング・コードを連続的に生成する。なお、参照符号21は、初期値発生器である。このような行列演算を利用した1サイクルで、1シンボル分のコードを発生させる技術については、本願出願人が先に提案している特願平11-48504号に詳細に記載されている。

【0041】また、スプレッディング・コード発生器11については、簡単な構成で、パラレルにコードを発生することを可能とするために特殊な構成を採用する。すなわち、スプレッディング・コードを構成する行列のうちの一部をテーブル化し、そのテーブルの値に簡単な論理演算を行なうことで、1サイクルで1シンボル長のコードをパラレルに発生させる。スプレッディング・コード発生器の詳細については、図4~図9を用いて、後に詳細に説明する。

【0042】(実施の形態2) 図2は、本発明の実施の形態2にかかる、マッチドフィルタを用いた初期同期装置の回路構成を示したものである。本実施の形態の特徴は、拡散符号発生器102からパラレルにコードを生成することである。これにより、シリアルにコードを発生させていた図11の従来例に比べ、処理時間を1/拡散率に短縮することができる。

【0043】なお、前掲の実施の形態(図1)では、スプレッディング・コードとスクランブリング・コードの各発生器を示していたが、スクランブリング・コード発生器については、図1(b)を用いて説明したため、説明の簡略化のため、本実施の形態(次の実施の形態(図3)でも同じ)では、スプレッディング・コード発生器のみを記載している。すなわち、図2中の拡散符号発生器102(および図3中の拡散符号発生器102)は、スプレッディング・コード発生器である。

【0044】本実施の形態の初期同期装置は、図示されるように、マッチドフィルタ101と拡散符号発生器102と演算用の拡散符号を蓄積する演算用レジスタ103で構成されている。

【0045】同図に示すマッチドフィルタ101はFIRデジタルフィルタを用いた4倍拡散のマッチドフィルタの例である。このマッチドフィルタ101は、直列接続された複数のフリップフロップ104~107で構成され、一段目のフリップフロップ104に受信ベースバンド信号が与えられ、フリップフロップ群104~107にクロックがパラレルに与えられるようにしているシフトレジスタ108と、シフトレジスタ108に蓄えられた受信ベースバンド信号と拡散符号列との逆拡散演算を実行する乗算器109~112、乗算結果を加算する加算器113と、を備えている。

【0046】拡散符号発生器102は、拡散符号列を2符号ずつパラレルに発生し、2クロック(拡散率/2のクロック数)で拡散符号列分の拡散符号を発生させる。

【0047】演算用レジスタ103は、直列接続された複数のフリップフロップ群114~117と直列接続された複数のフリップフロップ118~121の2段で構成され、1サイクル目には、フリップフロップ116、フリップフロップ117に拡散符号発生器102より与えられる拡散符号がパラレルに入力され、2サイクル目には、フリップフロップ114、フリップフロップ11

5に拡散符号発生器102より与えられる拡散符号がパラレルに入力され、フリップフロップ114~117にクロックがパラレルに与えられる。

【0048】フリップフロップ114~117に拡散符号列が格納されたあと、フリップフロップ114~117の値をフリップフロップ118~121にパラレルにロードする。また、フリップフロップ群118~121にクロックがパラレルに与えられるようにしている。

【0049】次に、図2の初期同期装置の動作について説明する。

【0050】受信ベースバンド信号は、クロックに同期した6ビットの信号である。デジタル信号は、フリップフロップ104に入力されたのち、フリップフロップ107に向けてクロックに同期してシフトされる。4クロック後には、フリップフロップ104~107にデータが蓄積され、フリップフロップ104~107にパラレルに与えられていたクロックは停止される。これにより、フリップフロップ104~107に蓄積された受信ベースバンド信号の値は保持される。

【0051】次に、拡散符号発生器102が2符号（拡散率の1/2）ずつパラレルに発生すると、まずパラレルに2符号出力された拡散符号列は、フリップフロップ群117~116にクロックに同期してパラレルに格納される。つぎに残りの2符号が拡散符号発生器よりパラレルに出力された拡散符号列は、フリップフロップ群115~114にパラレルに格納される。つまり、2クロック後には、フリップフロップ群114~117に拡散符号列は格納される。次のクロックで、フリップフロップ群114~117の値をフリップフロップ群118~121にロードする。以上により、受信ベースバンド信号との相関検出を行なう拡散符号列がセットされる。

【0052】乗算器109~112は、6ビット×1ビットの乗算器であり、7ビットの出力信号の出力を行い、加算器113は、7ビット+7ビット+7ビット+7ビットの加算器であり、9ビットの出力信号を出力する。乗算器109では、フリップフロップ104（6ビット）の出力信号とフリップフロップ118の出力信号との乗算が行われる。乗算器110~112では、フリップフロップ群105~107とフリップフロップ群119~121との乗算がそれぞれ行われる。加算器113では乗算器109~112の出力の加算が行われ、乗算器109~112、加算器113を用いて受信ベースバンド信号と拡散符号発生器102で発生する拡散符号列との相関結果を出力する。

【0053】次に、拡散符号列の切り換えの動作について説明する。

【0054】次の相関演算を行なう拡散符号列が、拡散符号発生器102より2符号ずつパラレルに出力されると、クロックに同期してフリップフロップ群118~121に3クロックで格納されるため、拡散符号列の切り

換えが3クロックで行なわれる。拡散符号の切り換えが完了後、シフトレジスタ108に蓄えられている受信ベースバンド信号と切り換え後の拡散符号列との相関検出が行なわれる。このような方式を採用すれば、複数の種類の拡散符号列の切り換えを連続して行う場合、相関検出を行なおうとしている拡散符号列の種類が10種類であれば、10種類の拡散符号列に対し30クロックで相関検出が行なわれる。本動作は、受信ベースバンド信号の拡散率が変わっても同様な動作で行なう事が可能である。

【0055】したがって、本実施の形態によれば、拡散符号列の切り換えの高速化が可能であり、初期同期で行なう拡散符号の検出及びタイミングの確立の高速化を図ることができる。

【0056】（実施の形態3）図3は、本発明の実施の形態3にかかる初期同期装置の構成を示すブロック図である。

【0057】本実施の形態の初期同期装置の特徴は、拡散符号発生器202が、4符号（4ビットの符号）をパラレルに出力することである。すなわち、1サイクルで、必要な拡散符号列が、拡散符号発生器202から出力される、これにより、演算用レジスタ203の値の切り換えが連続したサイクルで行なえる。その他の構成は、実施の形態1とほぼ同じである。

【0058】なお、図2に示す実施の形態2にかかる初期同期装置と同一の機能を有する部分には同一符号を付し説明の重複を避ける。

【0059】本実施の形態の初期同期装置には、拡散符号列を4符号パラレルに発生し、1クロック（拡散率/4のクロック数）で拡散符号列分の拡散符号を発生させる拡散符号発生器202が備えられている。

【0060】演算用レジスタ203は、直列接続された複数のフリップフロップ群214~217で構成され、1サイクルで、一段目フリップフロップ214、二段目のフリップフロップ215、三段目フリップフロップ216、四段目のフリップフロップ217に拡散符号発生器202より与えられる拡散符号がパラレルに入力される。また、フリップフロップ214~217にクロックがパラレルに与えられるようにしている。以上により、受信ベースバンド信号との相関検出を行なう拡散符号列がセットされる。

【0061】以上のように構成された、本実施の形態の拡散符号の切り換えの動作について説明する。

【0062】つぎの相関演算を行なう、拡散符号列が同様に拡散符号発生器202より4符号パラレルに出力されると、クロックに同期して演算用レジスタ203に1クロックで格納されるため、拡散符号列の切り換えが1クロックで行えられる。拡散符号の切り換えが完了後、シフトレジスタ108に蓄えられている受信ベースバンド信号と切り換え後の拡散符号列との相関検出が行なわ

れる。同様に、複数の種類の拡散符号列の切り換えを連続して行うとすると、相関検出を行なおうとしている拡散符号列の種類が10種類であれば、10種類の拡散符号列に対し10クロックで相関検出が行なわれる。

【0063】本動作は、受信ベースバンド信号の拡散率がかわっても同様な動作で行なう事が可能である。

【0064】したがって、本実施の形態によれば、拡散符号列の切り換えのさらなる高速化が可能であり、初期同期で行なう拡散符号の検出及びタイミングの確立の高速化を図ることが出来る。また、演算用レジスタを2段構成にせずに、相関検出の動作を止めることなく、拡散符号列の切り換えを行なうことができるため、装置の小型化も達成される。

【0065】（実施の形態4）本実施の形態では、主に、上述の実施の形態1～3で使用されるスペルディング・コード（spreading code）の発生器の具体的な構成と動作について説明する。

【0066】図4は、拡散符号発生器の構成を示すブロック図である。図示される拡散符号発生器300は、アダマール符号（Hadamard Code）発生器である。アダマール変換は、画像処理などに用いられる直交変換法の一様である。変換核は、+1と-1を要素とする正方形のマトリクス（これをアダマール行列という）であり、アダマール符号は、+1、-1の代わりに、0、1を用いたアダマール行列と、補行列により作られる線形の2進符号である。

【0067】図4のアダマール符号発生器300では、簡単な構成で拡散符号を平行に生成させることを可能とするために、3種類の参照用データA、B、Cが登録されているテーブル307と、ROM化されたワイヤードロジック301、302、303と、を設ける。3種類の参照用データA、B、Cは“1”と“0”の組み合わせであり、具体的には、例えば、“1”を電源電位とし、“0”をグランドとして、配線により予めプログラムされている。

【0068】そして、A～Cのうちのどの参照用データを用いるかを、端子310、311、312の各々に与える、アダマール番号を2進表記した場合の各ビットのデータ（“0”又は“1”）の組み合わせで制御する。

【0069】各端子310、311、312に与えるアダマール番号（2進表記）の組み合わせに応じて、アダマール符号発生器300からは、1シンボル長（ここでは8ビット）の、異なるアダマール符号が平行に出力される。なお、参照符号304は、8ビットの各々のビット毎に、排他的論理和をとる排他的論理和ゲートである。

【0070】そして、排他的論理和ゲート306により、並列に出力される8ビットのアダマール符号（スペルディング・コード）と、Golay符号発生器305から発生する8ビットの平行符号（スクリンプリング

・コード）との排他的論理和をとる（対応する各ビットについて同時に排他的論理和をとる）ことにより、第2段階用の符号が平行に生成されることになる。

【0071】以下、具体的に説明する。

【0072】図示されるように、拡散コード発生器300が具備する参照用データAは、“01010101”であり、参照用データBは、“00110011”であり、参照用データCは、“00001111”である。これらは、拡散コードを構成する行列のうちの数種類の一部をテーブル化したものである。

【0073】また、ワイヤードロジック301～303の各々は、アダマール番号と参照用データの論理積をとる複数のアンドゲートからなる。

【0074】次に、アダマール行列生成法について、具体的に説明する。図5は、アダマール行列の生成を行なう行列式である。

【0075】図5中、kの値は0以上の整数を用いる。図6には、図5で示した、生成法にもとづきk=3のときに生成したそれぞれのアダマール番号に対する拡散符号列を示す。図6からわかるように、k=3の時は、アダマール番号0～7の8種類の拡散符号列が生成される。

【0076】つまり、アダマール番号1〔二進表示：001〕の時は、参照用データAで示す“01010101”（すなわち、1符号ずつ0と1を交互にした8符号で構成された符号列）となる。

【0077】また、アダマール番号2〔二進表示：010〕の時は、参照用データBで示す“00110011”（すなわち、2符号ずつ0と1を交互にした8符号で構成された符号列）となる。

【0078】また、アダマール番号3〔二進表示：011〕の時は、“01100110”となる。この拡散符号列は、アダマール番号1の拡散符号列と、アダマール番号2の拡散符号列との和をとったものと等価である。すなわち、拡散符号列7番目は、参照用データAおよびBで示される“01010101”、“00110011”の各々の拡散符号列7番目の0と0の排他的論理和をとり0に、拡散符号列6番目は、“01010101”、“00110011”の各々の拡散符号列6番目の1と0の排他的論理和をとり1に、拡散符号列5番目は、“01010101”、“00110011”の各々の拡散符号列5番目の0と1の排他的論理和をとり1になり、以下同様に繰り返し、アダマール番号3の符号列となる。

【0079】アダマール番号4〔二進表示：100〕の時は、参照用データCで示す“00001111”（すなわち、4符号ずつ0と1を交互にした8符号で構成された符号列）となる。

【0080】アダマール番号5〔二進表示：101〕の時は、“01011010”となり、この拡散符号列

は、アダマール番号4の拡散符号列と、アダマール番号1の拡散符号列との和をとったものと等価である。すなわち、拡散符号列7番目は、参照用データAおよびCで示される“01010101”、“00001111”の各々の拡散符号列7番目の0と0の排他的論理和をとり0に、拡散符号列6番目は、“01010101”、“00001111”の各々の拡散符号列6番目の1と0の排他的論理和をとり1に、拡散符号列5番目は、“01010101”、“00001111”の各々の拡散符号列5番目の0と0の排他的論理和をとり0になり、以下、同様に繰り返して、アダマール番号5の符号列となる。

【0081】アダマール番号6〔2進表示：110〕の時は、“00111100”となり、この拡散符号列は、アダマール番号4の拡散符号列と、アダマール番号2の拡散符号列との和をとったものと等価である。すなわち、参照用データBとCで示される“00110011”、“00001111”の各々の拡散符号の排他的論理和をとったものである。

【0082】また、アダマール番号7〔2進表示：111〕の時は、“01101001”となる。この拡散符号列は、アダマール番号4の拡散符号列と、アダマール番号1の拡散符号列と、アダマール符号列2の拡散符号列の和をとったものと等価である。すなわち、参照用データA、B、Cで示される“01010101”、“00110011”、“00001111”の各々の拡散符号の排他的論理和をとったものが、それぞれの拡散符号列となる。

【0083】以上のことからわかるように、アダマール番号（十進表示）を2進表示して、0ビット目と参照用データAで示される“01010101”で論理和をとったものと、1ビット目と参照用データBで示される“00110011”で論理和をとったものと、2ビット目と参照用データCで示される“00001111”で論理和をとり、各々論理和をとった結果について排他的論理和をとった結果が、アダマール符号列となる。

【0084】次に、図4のアダマール符号発生器300の動作について説明する。

【0085】ワイヤードロジック301～303は、8ビット幅の2入力信号の論理積をとり、8ビットの出力を行う。排他的論理和304は、8ビット幅の3入力信号の排他的論理和演算を行なうものであり、8ビットの平行出力を行なう。

【0086】ワイヤードロジック301では、アダマール番号を2進表示したものの0ビット目のデータと参照用データAとの論理積演算が行なわれる。ワイヤードロジック302では、2進表示のアダマール番号の1ビット目のデータと参照用データBの論理積演算が行なわれる。ワイヤードロジック303では、2進表示のアダマール番号の2ビット目のデータと参照用データCの論理

積演算が行なわれる。排他的論理和ゲート304では、ワイヤードロジック301～303から出力される8ビットの信号について、各ビット毎に排他的論理和をとるものであり、これにより、8符号で構成されたアダマール符号列が平行に発生する。

【0087】以上の動作により、1サイクルで8符号で構成されたアダマール符号列を平行の生成することが可能である。本動作は、kの値がかわっても同様の動作を行なう事が可能である。

【0088】したがって、本実施の形態によれば、少数の参照用データとわずかな論理回路で、アダマール番号を端子310、311、312に入力するだけで、アダマール符号を用いた複数の種類の拡散符号列の平行生成を可能とする。よって、拡散符号列の生成の高速化が図られ、さらに、拡散符号の切り換えの高速化が可能である。これにより、2段階処理で行なうスプレッディング符号グループの同定およびフレームタイミングの確立の高速化を図ることができる。

【0089】（実施の形態5）本実施の形態では、主に、上述の実施の形態1～3で使用されるスプレッディング・コード（spreading code）の発生器の、他の例について説明する。本実施の形態では、アダマール符号の代わりに、階層化直交符号を使用する。

【0090】図7は、第3段階用拡散符号を発生する拡散符号発生器の構成を示すブロック図である。図示されるように、拡散符号発生器は、階層化直交符号発生器600、ゴールド符号（GOLD符号）発生器601、階層化直交符号とゴールド符号の排他的論理和をとる排他的論理和回路602から構成されている。

【0091】排他的論理和回路602からは、8ビットの第3段階用拡散符号が平行に出力される。なお、ゴールド符号は、基本的には、周期が等しい2種類のM系列を用意して、それらを加算して得られる符号系列である。

【0092】階層化符号発生器600は、前掲の実施の形態のアダマール符号発生器と同様の構成をしている。すなわち、3種類の参照データD、E、Fをテーブルとして用意すると共に、ワイヤードロジック（ワイヤードROM）603、604、605と、エクスクルージブノア回路606をもつ構成とし、2進表記の階層化直交番号の各ビットを端子610、611、612に入力し、入力される“1”、“0”の組み合わせによって、どの参照用データを使用するかを切り替えていくことにより、異なる種類の、8ビットの階層化直交符号を平行に連続して生成していくものである。

【0093】以下、具体的に説明する。

【0094】テーブルに用意される参照用データDは、“10101010”であり、参照用データEは、“11001100”であり、参照用データFは、“11110000”である。ワイヤードロジック

603～605は、端子610～612から入力される階層化直交番号の各ビットのデータと参照用データの各ビットとの論理積をとる8個のANDゲートを有する。また、エクスクルーシブノア回路606は、各ワイヤードロジック603、604、605からパラレルに出力される8ビットのデータについて、各ビット毎に一致判定を行なう。

【0095】まず、階層化直交行列生成法について説明する。図8は、階層化直交行列の生成を行なう行列式である。nの値は0以上の整数を用いる。図9には、図8で示した、生成法に基づきn=3のときに生成したそれぞれの階層化直交番号に対する拡散符号列を示す。図9より、n=3の時は、階層化直交番号0～7（十進表記）の8種類の拡散符号列が生成されることがわかる。

【0096】図7の回路を用いた階層化符号の発生方法の概略は、以下のとおりである。すなわち、まず、十進表記の階層化直交番号を2進表記に直し、その2進表記のデータのMSBとLSBとを入れ替え（交換）する。このようにして得られたデータの最上位ビットを図7の端子612に入力し、中間ビットを図7の端子611に入力し、最下位ビットを図7の端子610に入力する。これにより、図9に示すような8ビットの直交化階層化符号がパラレルに生成される。

【0097】以下、具体的に説明する。

【0098】つまり、図9に示すように、階層化直交番号1〔2進表記：001〕の時は、参照用データFで示す“11110000”（すなわち、4符号ずつ0と1を交互にした8符号で構成された符号列）となる。

【0099】階層化直交番号2〔2進表記：010〕の時は、参照用データEで示す“11001100”（2符号ずつ0と1を交互にした8符号で構成された符号列）となる。

【0100】階層化直交番号3〔2進表記：011〕の時は、“11000011”となりこれは、階層化符号番号2と4の拡散符号列を加算したものと等価である。すなわち、拡散符号列の7番目は、参照用データFとEで示される“11110000”、“11001100”の各々の拡散符号列6番目の1と1が一致しているかの演算を行い、この結果として“1”となる。同様に、拡散符号列6番目は、参照用データFとEで示される“11110000”、“11001100”の各々の拡散符号列6番目の1と1が一致しているかの演算を行って“1”になり、拡散符号列5番目は、“11110000”、“11001100”の各々の拡散符号列5番目の1と0が一致しているかの演算を行って“0”になり、以下、同様に繰り返して、階層化直交番号3の符号列が得られる。

【0101】階層化直交番号4〔2進表記：100〕の時は、参照用データDで示す“10101010”（1符号ずつ0と1を交互にした8符号で構成された符号

列）となる。

【0102】階層化直交番号5〔2進表記：101〕の時は、階層化直交番号4と1の符号列の和“10100101”となる。すなわち、拡散符号列7番目は、参照用データFとDで示される“11110000”、“10101010”の各々の拡散符号列7番目の1と1が一致しているかの演算を行い“1”になり、拡散符号列6番目は、“11110000”、“10101010”の各々の拡散符号列6番目の1と0が一致しているかの演算を行い“0”になり、拡散符号列5番目は、“11110000”、“10101010”の各々の拡散符号列5番目の1と1が一致しているかの演算を行い“1”になり、以下、同様に繰り返して、階層化直交番号5の符号列が得られる。

【0103】階層化直交番号6〔2進表記：110〕については、階層化直交番号1および2の符号列の和となる。すなわち、参照用データFとEで示される“11001100”、“10101010”の各々の拡散符号が一致しているかの演算を行ったものが、拡散符号列となる。

【0104】同様に、階層化番号7〔2進表記：111〕については、参照用データF、E、Dで示される“11110000”、“11001100”、“10101010”の各々の拡散符号が一致しているかの演算を行い、その結果として、それぞれの拡散符号列が得られる。

【0105】つまり、階層化直交番号を2進表示してLSBとMSBの位置を入れ替えて逆順にし（例えば、MSBが5で、以下順に4、3、2、1なら、MSBを1とし、以下順に2、3、4、5にする）、そのデータの2ビット目（最上位ビット）と参照用データFで示される“11110000”で論理積をとったものと、1ビット目と参照用データEで示される“11001100”とで論理積をとったものと、0ビット目（最下位ビット）と参照用データDで示される“10101010”で論理積をとり、各々論理和をとった結果が一致しているか演算した結果が、階層化直交符号列となる。

【0106】つぎに、図7の階層化符号発生器600の動作について説明する。

【0107】ワイヤードロジック603～605は、8ビット幅の2入力信号の論理積ゲートであり、8ビットの出力を行い、一致回路606は、8ビット幅の3入力信号の一致しているかの演算回路であり、8ビットの出力を行なう。ワイヤードロジック603では、階層化直交番号の2ビット目と参照用データDの論理積演算が行なわれる。ワイヤードロジック604では、階層化直交番号の1ビット目と参照用データEとの論理積演算が行なわれる。ワイヤードロジック605では、階層化直交番号の0ビット目と参照用データFとの論理積演算が行なわれる。

【0108】一致回路606では、各ワイヤードロジック603、604、605から出力される8ビットのデータの対応するビットの値が一致しているか否かの演算が行なわれ、その結果として、8符号で構成された階層化直交拡散符号列がパラレルに発生する。

【0109】以上の動作により、1サイクルで8符号で構成された階層化直交符号列をパラレルの生成することが可能である。つぎに、ゴールド符号発生器601より並列に出力される8ビットのゴールド符号列（スクランプリング・コード）と、階層化直交符号列（スプレッディング・コード）との排他的論理和をとる（排他的論理和回路602）ことにより、3段階用符号をパラレルに発生させることができる。

【0110】本動作は、 n の値がかわっても同様の動作を行なう事が可能である。したがって、本実施の形態によれば、少数の参照用データとわずかな論理回路を用いたROMに、階層化直交番号を入力するだけで、階層化直交符号を用いた複数の種類の拡散符号列のパラレル生成を実現できる。よって、拡散符号列の生成の高速化、さらに、切り換えの高速化が可能であり、3段階処理で行なうスプレッディング符号の同定の高速化を図ることができる。

【0111】（実施の形態6）図10は、本発明の初期同期装置を適用したCDMA受信装置（CDMA通信装置）の構成を示すブロック図である。

【0112】図示されるように、CDMA受信装置は、受信アンテナ901と、所定の周波数でフィルタリング及び増幅する高周波信号処理部902と、アナログ信号をデジタル信号に変換するAD変換器903と、受信信号を復調するデータ復調部904と、復号を行なうデータ復号部905と、復号された信号を音声に換えるCODEC部906と、通信を行うものと同期の獲得を行なう初期同期装置907と、クロック信号部908と、タイミングコントロール部909と、を備えている。

【0113】初期同期装置907には、AD変換部903から拡散変調されたデジタル信号からなる受信ベースバンド信号が与えられ、クロック発生部908から与えられるクロックが印加される。タイミングコントロール部909では、逆拡散を行なうタイミング等の制御を行なう。初期同期装置907では、AD変換部903から与えられる受信ベースバンド信号との逆拡散演算が行なわれ、同期獲得をした結果として逆拡散演算結果がデータ復調部904に出力される。データ復調部904では、初期同期装置907より得られた逆拡散演算結果をもとにデータ復調し、データ復号部905にデータを与える。

【0114】なお、初期同期装置907は、前掲の実施の形態で説明したものと同一構成を有しており、拡散符号列をパラレルに連続サイクルで生成することが可能であるため、初期同期獲得の高速化を図ることができる。

このように、本発明のCDMA受信装置、初期同期獲得の高速化を図ることができる。

【0115】また、前掲の実施の形態で説明した初期同期装置のいずれかを、CDMA方式の移動体無線通信を行う基地局装置または移動体装置の無線受信部に備えてもよいし、その他の通信端末の無線受信部として使用してもよい。この場合も、初期同期の早期獲得に有効である。

【0116】

【発明の効果】以上説明したように、本発明は、拡散符号列をパラレルに連続サイクルで生成を行なうことにより、複数の種類の拡散符号列との相関検出の高速化を実現できる。これにより、初期同期の早期確立を行うことができ、システムの高性能化を図ることができる。

【図面の簡単な説明】

【図1】（a）本発明の実施の形態1にかかる初期同期装置の構成を示すブロック図

（b）スクランプリング・コード発生器がパラレルデータを生成する原理を説明するための図

【図2】本発明の実施の形態2における初期同期装置の構成を示すブロック図

【図3】本発明の実施の形態3における初期同期装置の構成を示すブロック図

【図4】本発明の実施の形態4にかかる拡散符号発生器の構成を示すブロック図

【図5】実施の形態4における、拡散符号列（アダマール符号）の生成方法を説明するための図

【図6】図4の回路における、アダマール番号と発生する拡散符号列との関係を示す図

【図7】本発明の実施の形態5にかかる拡散符号発生器の構成を示すブロック図

【図8】本発明の実施の形態5における拡散符号列（階層化直交符号）の生成方法を説明するための図

【図9】本発明の実施の形態5における、階層化直交番号と発生する拡散符号列との関係を示す図

【図10】本発明の実施の形態6にかかるCDMA受信装置の構成を示すブロック図

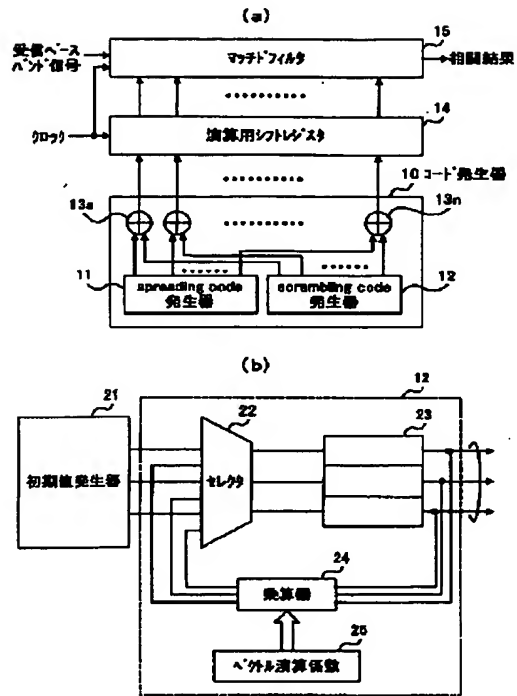
【図11】従来の初期同期装置の一例の構成を示すブロック図

【符号の説明】

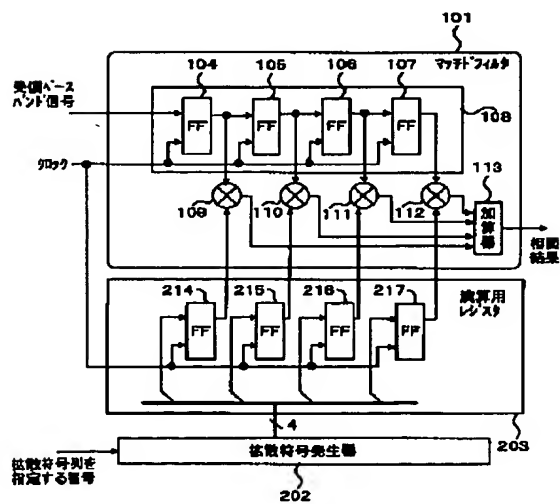
- 10 コード発生器
- 11 スプレッディング・コード発生器
- 12 スクランプリング・コード発生器
- 13 a ~ 13 n 加算器
- 14 演算用シフトレジスタ
- 15 マッチドフィルタ
- 21 初期値発生器
- 22 セレクタ
- 23 並列入出力のシフトレジスタ
- 24 乗算器

2.5 ベクトル演算係数

【図1】



【図3】

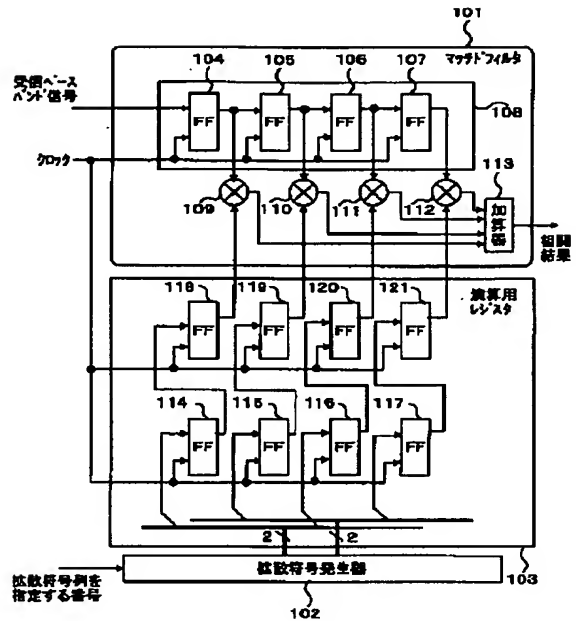


【図5】

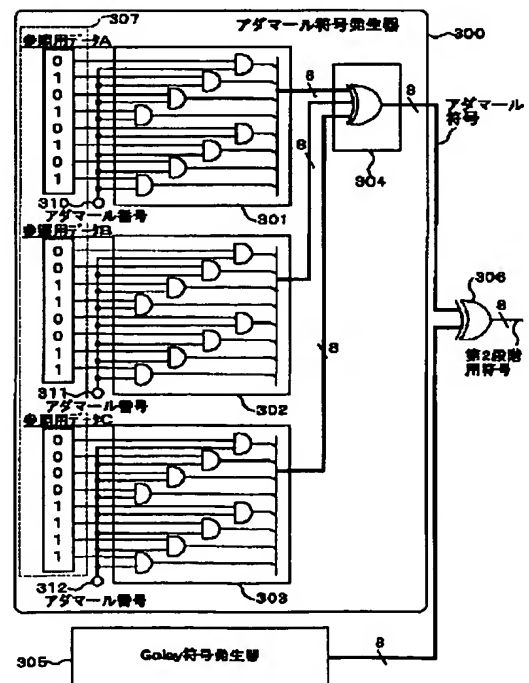
$$H_0 = 0$$

$$H_k = \begin{pmatrix} H_{k-1} & H_{k-1} \\ H_{k-1} & H_{k-1} \end{pmatrix}$$

【図2】



【図4】



【図6】

7ビットのデコード番号 (2進表示)	拡張符号列
0 [0 0 0]	0 0 0 0 0 0 0 0 ← 0
1 [0 0 1]	0 1 0 1 0 1 0 1 ← 1(参照用デコード)
2 [0 1 0]	0 0 1 1 0 0 1 1 ← 2(参照用デコード)
3 [0 1 1]	0 1 1 0 0 1 1 0 ← 1+2
4 [1 0 0]	0 0 0 0 1 1 1 1 ← 4(参照用デコード)
5 [1 0 1]	0 1 0 1 1 0 1 0 ← 4+1
6 [1 1 0]	0 0 1 1 1 1 0 0 ← 4+2
7 [1 1 1]	0 1 1 0 1 0 0 1 ← 4+1+2

【図8】

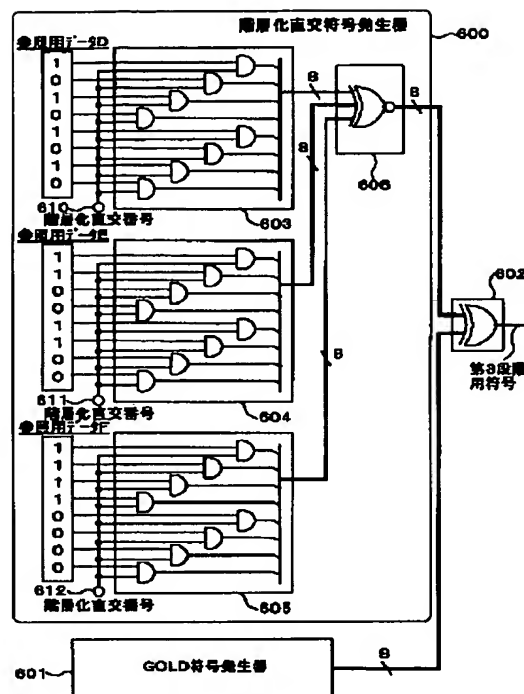
$$C_0(0) = 1$$

$$\begin{pmatrix} C_{n+1}(0) \\ C_{n+1}(1) \\ C_{n+1}(2) \\ C_{n+1}(3) \\ \vdots \\ C_{n+1}(2^{n+1}-1) \end{pmatrix} = \begin{pmatrix} C_n(0) & \overline{C_n(0)} \\ C_n(0) & \overline{C_n(0)} \\ C_n(1) & \overline{C_n(1)} \\ C_n(1) & \overline{C_n(1)} \\ \vdots & \vdots \\ C_n(2^n-1) & \overline{C_n(2^n-1)} \end{pmatrix}$$

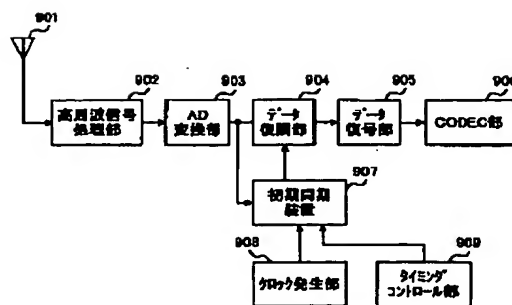
【図9】

7ビットのデコード番号 (2進表示)	拡張符号列
0 000	1 1 1 1 1 1 1 1 ← 0
1 [001](100)	1 1 1 1 0 0 0 0 ← 4(参照用デコード)
2 010	1 1 0 0 1 1 0 0 ← 2(参照用デコード)
3 [011](110)	1 1 0 0 0 0 1 1 ← 2+4
4 [100](001)	1 0 1 0 1 0 1 0 ← 1(参照用デコード)
5 101	1 0 1 0 0 1 0 1 ← 4+1
6 [110](011)	1 0 0 1 1 0 0 1 ← 1+2
7 111	1 0 0 1 0 1 1 0 ← 1+2+4

【図7】



【図10】



【図 11】

